Fabric-CA 操作指南

一. Fabric-ca 工具的使用

1 . Fabric-ca 服务的搭建

CA 服务是为了联盟中的节点或者用户颁发身份证书。Fabric 中提供了两种方法生成证书：
fabric-ca-server/fabric-ca-client
cryptogen
其中 fabric-ca-server/fabric-ca-client 分别是用来搭建 CA 服务和与之交互的工具。而 cryptogen 是用来根据配置文件生成证书的工具。

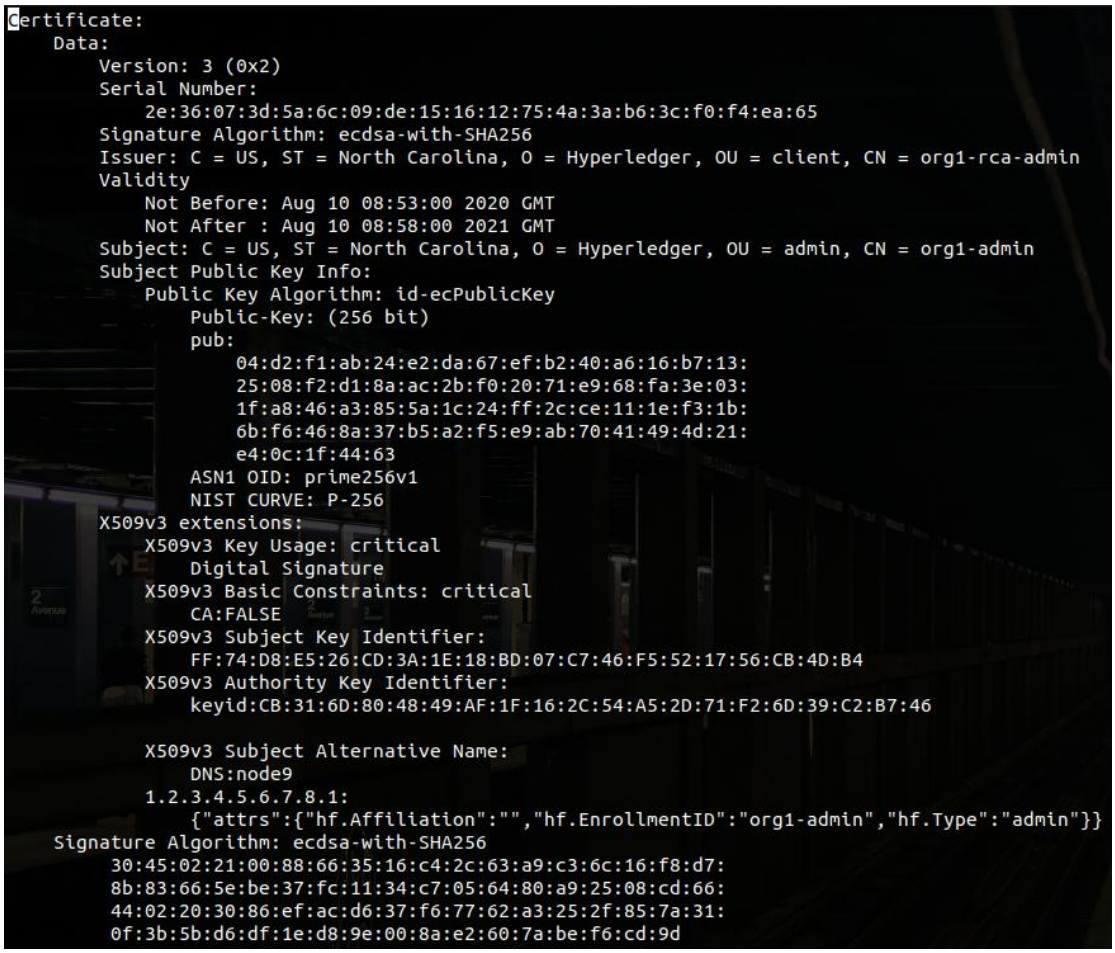
但是 fabric 文档中提到了 cryptogen 只适用于测试，在实际的生产网络中一般不会使用。[https://hyperledger-fabric.readthedocs.io/en/release-2.0/commands/cryptogen.html#cryptogen]

证书颁发的流程

CA（Cetificate Authority）证书颁发的流程：
1.节点在本地生成一对公私钥，并将公钥与自己的身份发送给 CA
2.CA 在接收到后用自己的私钥对其签名，并附上一些相关的字段，以生成.pem 证书
3.然后 CA 将.pem 证书发送给节点

证书的具体内容

证书的内容可由下图表示



其中每个字段的含义

Version： 版本号。说明当前证书的版本，不同版本，可能会出现不同字段

Serial Number： 序列号。 一个 CA 颁发的证书会有一个唯一的序列号

Signature Algorithm：CA 签发这张证书使用的签名算法。

Issuer：签发证书的 CA 的相关信息

C：国名

ST：州/省

O：机构名

OU：机构单元名称

CN：通用名

Validity：有效期 标明证书的有效时间

Subject： 主体名。标识证书主体的信息

C：

ST：

O：

OU：

CN：

Subject Public Key Info： 主体公钥信息
Public Key Algorithm：
Pub：
ASN1 OID：
NIST CURVE：
V3 版本的扩展字段
X509v3 extensions

搭建 CA 服务器

Fabric 提供了一个搭建 CA 服务器的工具 fabric-ca-server，它的所有命令可以参照如下链接：
[https://hyperledger-fabric-ca.readthedocs.io/en/release-1.4/servercli.html#fabric-ca-server-s-cli]
Fabric-ca-server 提供了三条命令
init: 初始化 fabric-ca 服务
start: 启动 fabric-ca 服务
version：显示 Fabric CA 服务的版本号

Fabric CA 服务搭建分三步： 初始化（init）-> 修改配置文件 -> 启动服务（start）-> 生成 admin 的证书。对 Fabric CA server 具体的配置的设定可以通过三个方式
1.命令参数
2.环境变量
3.配置文件
（前者的设定可以覆盖后者的设定，对 Fabric ca server 配置的具体细节，我们以配置文件来讲解。）

fabric-ca-server 的家目录（即工作目录也有如下判定规则）
1.以命令行中-home 提供的参数为家目录，若未指定则使用 2
2.以 FABRIC_CA_SERVER_HOME 为家目录，若未指定则使用 3
3.以 FABRIC_CA_HOME 为家目录，若未指定则使用 4
4.以 CA_CFG_PATH 为家目录，若未指定则使用 5
5.当前工作路径

预备阶段: 建立 fabric-ca 的工作目录，并指定家目录
ca-server:~$ mkdir ca-server
ca-server:~$ cp ./bin/fabric-ca-server ca-server
初始化（init）
ca-server:~/ca-server$ ./fabric-ca-server init -b admin:adminpw
当禁用 LDAP 时，需要-b（bootsrtap identity）提供 fabric-ca-server 的引导用户。这个引导用户同时也是 fabric-ca-server 的 administrator。运行成功时，会有如下 log。

```
node@node9:~/ca-server$ ./fabric-ca-server init -b admin:adminpw

2020/09/03 16:40:40 [INFO] Created default configuration file at /home/node/ca-server/fabric-ca-server-config.yaml
2020/09/03 16:40:40 [INFO] Server Version: 1.4.7
2020/09/03 16:40:40 [INFO] Server Levels: &{Identity:2 Affiliation:1 Certificate:1 Credential:1 RAInfo:1 Nonce:1}
2020/09/03 16:40:40 [WARNING] &{69 The specified CA certificate file /home/node/ca-server/ca-cert.pem does not exist}
2020/09/03 16:40:40 [INFO] generating key: &{A:ecdsa S:256}
2020/09/03 16:40:40 [INFO] encoded CSR
2020/09/03 16:40:40 [INFO] signed certificate with serial number 173480065115608509451883209757188476268323421422
2020/09/03 16:40:40 [INFO] The CA key and certificate were generated for CA
2020/09/03 16:40:40 [INFO] The key was stored by BCCSP provider 'SW'
2020/09/03 16:40:40 [INFO] The certificate is at: /home/node/ca-server/ca-cert.pem
2020/09/03 16:40:41 [INFO] Initialized sqlite3 database at /home/node/ca-server/fabric-ca-server.db
2020/09/03 16:40:41 [INFO] The issuer key was successfully stored. The public key is at: /home/node/ca-server/IssuerPublicKey, secret key is at: /home/node/ca-server/msp/keystore/IssuerSecretKey
2020/09/03 16:40:41 [INFO] Idemix issuer revocation public and secret keys were generated for CA ''
2020/09/03 16:40:41 [INFO] The revocation key was successfully stored. The public key is at: /home/node/ca-server/IssuerRevocationPublicKey, private key is at: /home/node/ca-server/msp/keystore/IssuerRevocationPrivateKey
2020/09/03 16:40:41 [INFO] Home directory for default CA: /home/node/ca-server
2020/09/03 16:40:41 [INFO] Initialization was successful
```

Init 阶段：

init 成功时会在家目录下产生如下文件：

```
node@node9:~/ca-server$ tree
.
├── ca-cert.pem
├── fabric-ca-server
├── fabric-ca-server-config.yaml
├── fabric-ca-server.db
├── IssuerPublicKey
├── IssuerRevocationPublicKey
└── msp
    └── keystore
        ├── ab5c5afe84d69f2d14e739ec8f0650d96e2e6e1154e30650c48b67e4fdf9a119_sk
        ├── IssuerRevocationPrivateKey
        └── IssuerSecretKey

2 directories, 9 files
```

其中

ca-cert.pem: #自签名的证书

fabric-ca-server-config.yaml: 配置文件

fabric-ca-server.db

IssuerPublicKey

IssuerRevocationPublicKey

msp

Keystore

ab5*_sk

IssuerRevocationPrivateKey

IssuerSecretKey

修改配置文件
参照
https://hyperledger-fabric-ca.readthedocs.io/en/latest/deployguide/cadeploy.html

启动 CA 服务
由于 CA 自签名的证书内容与.yaml 配置文件相关，所以当修改了.yaml 文件之后，需要删除已产生的 MSP 文件夹和 ca-cert.pem 证书，然后再启动服务。启动服务时，fabric-ca-server 会自动检测当前家目录下是否存在私钥和证书。如果不存在会先根据.yaml 文件生成私钥和证书然后再启动服务。

```
node@node9:~/ca-server$ ./fabric-ca-server start

2020/09/04 11:36:47 [INFO] Configuration file location: /home/node/ca-server/fabric-ca-server-config.ya
ml
2020/09/04 11:36:47 [INFO] Starting server in home directory: /home/node/ca-server
2020/09/04 11:36:47 [INFO] Server Version: 1.4.7
2020/09/04 11:36:47 [INFO] Server Levels: &{Identity:2 Affiliation:1 Certificate:1 Credential:1 RAInfo:
1 Nonce:1}
2020/09/04 11:36:47 [WARNING] &{69 The specified CA certificate file /home/node/ca-server/ca-cert.pem d
oes not exist}
2020/09/04 11:36:47 [INFO] generating key: &{A:ecdsa S:256}
2020/09/04 11:36:47 [INFO] encoded CSR
2020/09/04 11:36:47 [INFO] signed certificate with serial number 492091247330674221155556390184944970685
762884209
2020/09/04 11:36:47 [INFO] The CA key and certificate were generated for CA test2.ca
2020/09/04 11:36:47 [INFO] The key was stored by BCCSP provider 'SW'
2020/09/04 11:36:47 [INFO] The certificate is at: /home/node/ca-server/ca-cert.pem
2020/09/04 11:36:47 [INFO] Initialized sqlite3 database at /home/node/ca-server/fabric-ca-server.db
2020/09/04 11:36:47 [INFO] The issuer key was successfully stored. The public key is at: /home/node/ca-
server/IssuerPublicKey, secret key is at: /home/node/ca-server/msp/keystore/IssuerSecretKey
2020/09/04 11:36:47 [INFO] Idemix issuer revocation public and secret keys were generated for CA 'test2
.ca'
2020/09/04 11:36:47 [INFO] The revocation key was successfully stored. The public key is at: /home/node
/ca-server/IssuerRevocationPublicKey, private key is at: /home/node/ca-server/msp/keystore/IssuerRevoca
tionPrivateKey
2020/09/04 11:36:47 [INFO] Home directory for default CA: /home/node/ca-server
2020/09/04 11:36:47 [INFO] Operation Server Listening on 127.0.0.1:9444
2020/09/04 11:36:47 [INFO] Listening on http://0.0.0.0:7059
```

注： 在 fabric-ca-server 初始化时，我们给了一个 admin 的账户。之后如果需要通过 ca 服务颁发证书，需要先进行注册（register）操作，将实体的身份以账户密码的形式提交给 ca 服务，然后通过 enroll 操作完整证书的颁发。而 register 这一步的操作是需要以 admin 的身份进行的。这个过程有点类似于：一个实体将自己身份先发送给 ca 服务进行检验，而检验是否通过是由 ca 服务的 admin 完成的，这一步完成之后在进行公钥验证颁发证书的操作。
而以 admin 的身份进行 register 操作时，需要提供 admin 能够用自己的私钥进行类似签名的操作。因此，对于 admin 这个账户，已经启动的 ca 服务需要能够为其颁发一个证书。这个过程需要存储 admin 公私钥的机器上先生成一对公私钥，然后公钥发送给 ca 服务，获得证书。（这一步操做也是通过 fabric-ca-client 完成的）

2．Fabric-客户端如何注册身份，申请证书

fabric-ca-client 工具用来进行身份的注册，其具体的用法可参照链接：
[https://hyperledger-fabric-ca.readthedocs.io/en/release-1.4/clientcli.html#fabric-ca-client-s-cli]
它的家目录的判断有如下规则：
1.以命令行中-home 指定的为家目录，若未指定则使用 2
2.以 FABRIC_CA_CLIENT_HOME 指定的为家目录，若未指定则使用 3
3.以 FABRIC_CA_HOME 指定的为家目录，若未指定则使用 4
4.以 CA_CFG_PATH 指定的为家目录，若未指定则使用 5
5.以${HOME}/.fabric-ca-client 为家目录

有 fabric-ca-client 与 CA 服务交互得到身份证书，分为两步，这两步之间的关系是：
1. 用户先将自己的身份信息，如组织所在国家，城市，机构，名称等以账号密码的形式，注册保存到 CA 服务上（register）
2. 用户在本地生成一对公私钥，提出证书请求，将公钥发送给 CA，同时需要提供之前注册账号的账户和密码（enroll）

其中第一步操作需要以 CA 服务 admin 的身份去执行，即通过--mspdir 参数指定 admin 的 msp 路径。
具体操作是：

```
~/ca-client$ fabric-ca-client register \
                --id.name username \
                --id.secret userpasswd \
                --id.type admin \
                -u http://ca-server-host:port \
                --mspdir ./ca-server-admin/msp
```

然后第二步执行 enroll 操作：

```
~/ca-client$ fabric-ca-client enroll \
                -u http://username:userpasswd@ca-server-host:port \
                --mspdir ./user/msp
```

运行结果如下：